To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

32176 Group

Combination of DMAC and Serial Interface Transmission

1. Overview

The reference sample program combined DMAC and serial interface transmission for 32176 group appears on this document.

2. Introduction

These application examples in this document are used in the following microcomputers and conditions.

- Microcomputer: 32176 Group (M32176FnVFP, M32176FnTFP)
- Operating frequency: 20 to 40 MHz (The sample program is compiled assuming a frequency of 40 MHz.)
- Operating Board: Starter kit for 32176 Group

3. Sample program for Combination of DMAC and Serial Interface Transmission

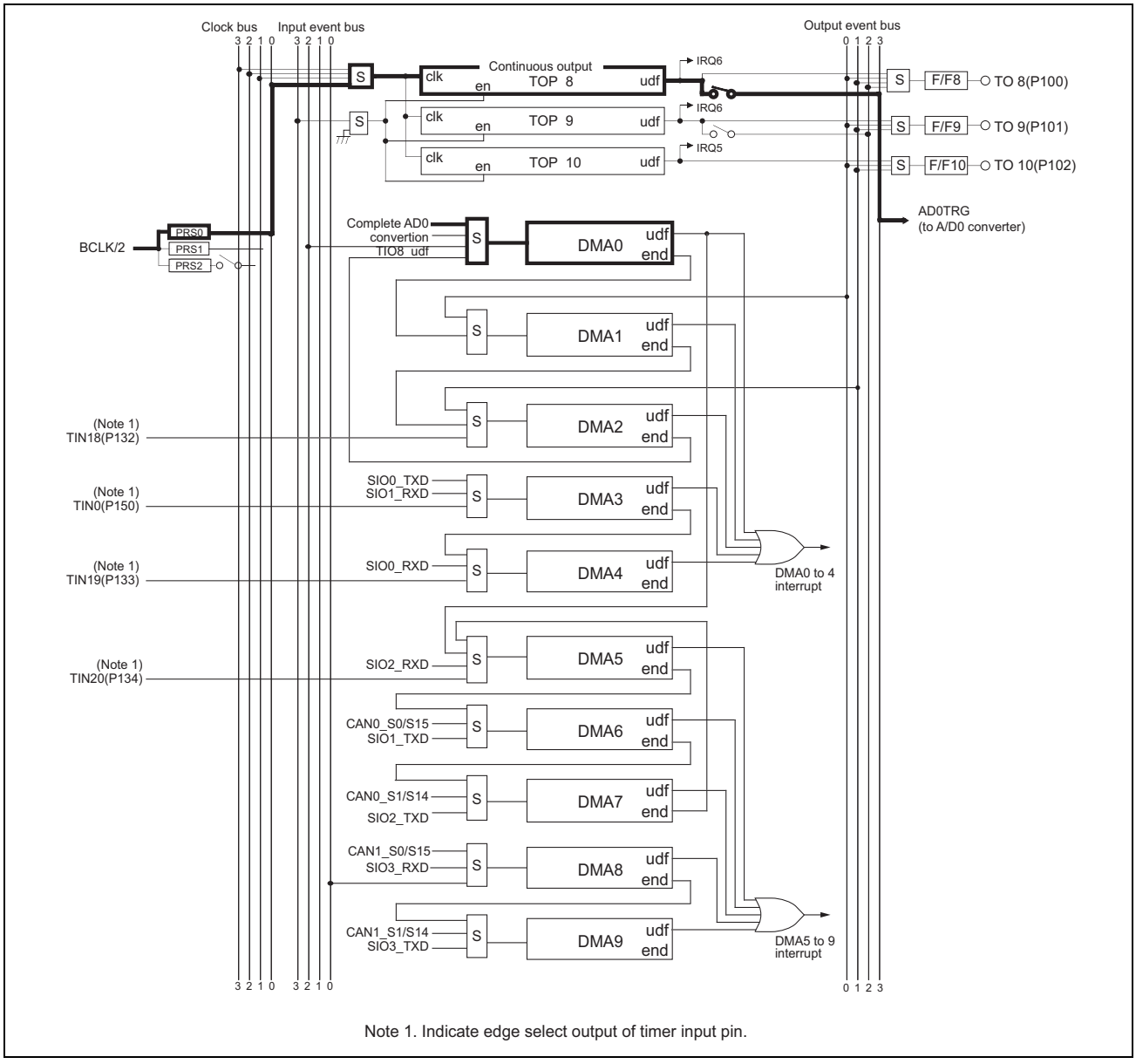
3.1 Outline of Sample program

In this sample program, the conversion result of A/D0 is automatically transmitted by the serial interface with using DMA.

During A/D converting the four channel from 0 to 3 of A/D0 converter in continuation scan mode, A/D conversion of channel0 is started at a fixed interval in single mode. DMA0 is started by completing A/D conversion in that single mode, automatic serial transmission is operated by transmit A/D converting result to transmisstion buffer of serial interface 0. A/D conversion in single mode is operated by operating timer TOP8 in continuous mode and automatically started at that underflow timing. Also A/D conversion result to serial transmission buffer is transmitted infinite times using DMA0 as source and using fixed distination ring buffer mode.

All processing above are operated without software load.

During program operation, value of port 13 is read out, and A/D converting result in continuous scan mode which is in 0 to 3 channel is outputted continuously to port11 by value of port 13.



Note 1. Indicate edge select output of timer input pin.

Figure3.1.1 Configuration of Sample for Combination of serial interface transmission and DMAC

3.2 Description of a reference program

Note. The registers used are indicated as (register name: bit name).

3.2.1 Various initialization functions (init_func())

(1) Call the port initialization function
(2) Call the DMAC initialization function
(3) Call the serial interface initialization function
(4) Call the A/D converter initialization function
(5) Call the timer initialization function

3.2.2 Port initialization functions (port_init())

(1) Initial setting of input and output port

- Set Port Input Permit bit of Port Input Special Function Control Register as permit inputting (PICNT: PIEN0)
- Initialize P11 Data Register (P11DATA)
- Set P11 Direction Register as output mode. (P11DIR)
- Set P11 Operation Mode Register as input/output port. (P11MOD)

Note: If a Direction Register is set as output before setting up a Data Register, an unfixed value is outputted until writing will be performed to a Data Register.

(2) Initial setting of input pin

- Set P13 direction register as input mode. (P13DIR)
- Set P13 operation mode register as input/output port. (P13MOD)

3.2.3 DMAC initialization Function (dma_init())

(1) Interrupt Setting of DMA0

- Set DMA0 interrupt disable. (DM04ITMK: DMITMK0)
- Clear DMA0 interrupt request flag. (DM04ITMK: DMITST0)

(2) Address Setting of DMA0

- Set data address of A/D0 converter channel 0 to DMA0 source address. (DM0SA)
- Set SIO0 transmit buffer address to DMA0 destination address. (DM0DA)

(3) Set DMA0 channel control register (DM0CNT)

- Set ring buffer mode to DMA0 transmit mode.
- Clear DMA0 transfer request flag.
- Select A/D0 converting completion to DMA0 transfer request factor.
- Set DMA0 transfer as enable.
- Set DMA0 transfer size to 8 bits.
- Set address fixed to DMA0 source address.
- Set address fixed to DMA0 destination address.

3.2.4 Serial interface initialization Function (sio0tr_init())

(1) Setting transfer control (S0TCNT: CDIV, TEN)
- Set baud rate generator count source to f(BCLK)
- Set transmission paused.

(2) Set port
- Set port P82 to TXD0, port P83 to RXD0.

(3) Set transfer data format (S0MOD, S0BAUR)
- Set to 8-bit UART.
- Set 1 stop bit to stop length.
- Set to no parity.
- Set baud rate as 19200bps. (At 19200 bps when CPU clock is 40 MHz.)

(4) Interrupt setting
- Set as SIO0 transmit interrupt disable. (ISIO0TXCR: ILEVEL)
- Set as SIO0 receive complete interrupt. (SI03SEL: ISR0)
- Set as SIO0 transmit interrupt request disable. (SI03MASK: T0MASK)

(5) Start transmission
- Set SIO0 to transmission enable (S0TCNT: TEN)

3.2.5 A/D converter initialization Function (ad_init())

(1) Interrupt Setting
- Set A/D0 interrupt disable. (IAD0CCR: ILEVEL)

(2) Set A/D converter (single mode) (AD0SIM0, AD0SIM1)
- Set to hardware trigger of output event bus 3.
- Set DMA transfer request when complete A/D conversion.
- Set A/D conversion mode as A/D conversion
- Set A/D conversion speed as fast.
- Set as disable Sample & Hold.
- Set A/D analog input pin to ADiIN0.

(3) Set A/D converter (scan mode) (AD0SCM0, AD0SCM1)
- Set to hardware trigger of output event bus 3.
- Set scan mode to continuous mode.
- Set interrupt request when complete A/D conversion.
- Set A/D conversion speed as fast.
- Set as disable Sample & Hold.
- Set 0 to channel 3 for A/D scan loop.

(4) Start converting
- Start A/D0 scan mode converting. (AD0SCM0: ADCSTT)

3.2.6 Timer initialization Function (timer_init())

(1) Initial Setting of output pin

- Set port P100 operation mode bit of P10 operation mode register to input/output port. (P10MOD: P100MOD)

(2) Interrupt setting

- Set TOP8 output interrupt as disable. (TOPIR3: TOPIM8)

(3) Set timer TOP8

- Set TOP8 output for inputting output event bus 3. (OEBCR: OEB3S)
- Set TOP8 as continuous output mode, clock source as clock bus0. (TOP810CR: TOP8M, TOP810CKS)
- Set "prescaler divided by value -1" to prescaler register 0. (10us@10MHz)
- Set TOP8 counter. (TOP8CT)
- Set TOP8 reload register (TOP8RL).
- Set TOP8 Enable protect bit as permit rewriting (TOPPRO: TOP8PRO) (Note 1)
- Start TOP8 counting (TOPCEN)

Note 1. Set enable only for bit used, set disable for other bit.

3.2.7 Main Function (main())

(1) Call the interrupt prohibitive function
(2) Call the various initialization functions
(3) Call the interrupt permit function
(4) Infinite loop outputs to port P11 A/D converting result of channel corresponded to input status of port P13.

3.3 Reference sample program

The sample program for combination of DMAC and serial interface transmission is shown below.

Note that the sample program below requires the SFR definition file. The latest SFR definition file can be downloaded from Renesas Technology website. When using the SFR definitions file, adjust the path setting to match the operating computer environment.

3.3.1 dma_sio_tr.c

```
/*  FILE COMMENT   ************************************************
 *      M32R C Programming              Rev. 1.01
 *              < Sample Program for 32176 >
 *              < AD-DMAC-SIO >
 *
 *      Copyright (c) 2004 Renesas Technology Corporation
 *                         All Rights Reserved
 ************************************************************************/

/************************************************************************/
/*              Include file                                            */
/************************************************************************/

#include                "..\inc\sfr32176_pragma.h"

/************************************************************************/
/*              Definition of external reference                        */
/************************************************************************/

extern  void            DisInt( void );                 /* Interrupt disable function */
extern  void            EnInt( void );                  /* Interrupt enable function */

/************************************************************************/
/*              Function prototype declaration                          */
/************************************************************************/

        void            main(void);                     /* Main function */
        void            init_func(void);                /* Initial setup function */
        void            port_init(void);                /* Initialize port  */
        void            timer_init(void);               /* Timer initialization */
        void            dma_init(void);                 /* Initialize DMA */
        void            sio0tr_init(void);              /* Initialize serial I/O */
        void            ad_init(void);                  /* Initialize A-D converter */

/************************************************************************/
/*              Define macro                                            */
/************************************************************************/

/* DMAC setting data */

                                                /* 0123 4567                                    */
#define DMA0_INIT       0x9c                    /* 1001 1100B   DMA0 channel control register   */
                                                /* |||| |||+--- Destination address fixed      */
                                                /* |||| ||+---- Source address fixed           */
                                                /* |||| |+----- Transfer size of 8-bit         */
                                                /* |||| +------ Transfer enabled              */
                                                /* ||++-------- 01: Start upon completion of A-D
conversion */
                                                /* |+---------- No transfer request           */
                                                /* +----------- Ring buffer mode               */

/* Serial I/O setting data */

                                                /* 0123 4567                                    */
#define P8MOD_SCI0      0x30u                   /* 0011 0000B   P8 operation mode register      */
                                                /* |||| |||+--- P87                            */
                                                /* |||| ||+---- P86                            */
                                                /* |||| |+----- P85                            */
                                                /* |||| +------ P84                            */
                                                /* |||+-------- RXD0                           */
                                                /* ||+--------- TXD0                           */
```

```
                                            /* ++---------- don't care                          */
                                            /* 0123 4567                                        */
#define SnTCNT_INI      0x00                /* 0000 0000B   SIOn transmit control register      */
                                            /* |||| |||+--- Disable transmission               */
                                            /* |||| +++---- don't care                          */
                                            /* ||++-------- f(BCLK)                             */
                                            /* ++---------- don't care                          */

#define SnMOD_INI       0x20                /* 0010 0000B   SIOn mode register                  */
                                            /* |||| |||+--- Sleep function disabled            */
                                            /* |||| ||+---- Parity inhibited                   */
                                            /* |||| |+----- don't care(odd)                     */
                                            /* |||| +------ 1 stop bit                         */
                                            /* |||+-------- Internal clock                      */
                                            /* +++--------- 8 bit UART                         */

/* Setting baud rate (Be sure to check actually set value when using) */

#define XIN             10                  /* 10MHz */
#define BAUD_19200      (XIN * 2000000 / 16 / 19200 - 1)        /* 19200bps */

/* A-D converter setting data */

                                            /* 0123 4567                                        */
#define AD0SIM0_INIT    0x98                /* 1001 1000B   AD0 single mode register 0          */
                                            /* |||| |||+--- AD conversion Start                 */
                                            /* |||| ||+---- AD conversion Stop                  */
                                            /* |||| |+----- Writing invalid                     */
                                            /* |||| +------ DMA transfer request               */
                                            /* |||+-------- Hardware trigger                    */
                                            /* ||+--------- Start output event bus 3            */
                                            /* |+---------- don't care                          */
                                            /* +----------- Start output event bus 3            */

#define AD0SIM1_INIT    0x40                /* 0100 0000B   AD0 single mode register 1          */
                                            /* |||| ++++--- ch0 selected                        */
                                            /* |||+-------- Sample & Hold speed                 */
                                            /* ||+--------- Sample & Hold enable               */
                                            /* |+---------- Double speed                        */
                                            /* +----------- AD conversion mode                 */

#define AD0SCM0_INIT    0x60                /* 0110 0000B   AD0 scan mode register 0            */
                                            /* |||| |||+--- AD conversion Start                 */
                                            /* |||| ||+---- AD conversion Stop                  */
                                            /* |||| |+----- Writing invalid                     */
                                            /* |||| +------ A-D0 interrupt request             */
                                            /* |||+-------- Hardware trigger                    */
                                            /* ||+--------- Start output event bus 3            */
                                            /* |+---------- Continuous mode                     */
                                            /* +----------- Start output event bus 3            */

#define AD0SCM1_INIT    0x44                /* 0100 0100B   AD0 scan mode register 1            */
                                            /* |||| +++---- 4ch scan                            */
                                            /* |||+-------- Sample & Hold speed                 */
                                            /* ||+--------- Sample & Hold enable               */
                                            /* |+---------- Double speed mode                   */
                                            /* +----------- don't care                          */

/* MJT setting data */

                                            /* 0123 4567                                        */
#define OEB3TOP8        0xC0u               /* 1100 0000B   Output event bus control register   */
                                            /* ||++ ++++---- don't care                          */
                                            /* ++---------- Select output event bus 3 for input:
TOP8 for output    */

                                            /* 0123 4567 89AB CDEF                              */
#define TOP8_MASK       0x0313u             /* 0000 0011 0001 0011B TOP8-10control register     */
#define TOP8_CntOutput  0x0210u             /* 0000 0010 0001 0000B                             */
                                            /*        ||    |   ++- Select clock bus 0          */
                                            /*        ||    +------ Select enable source       */
                                            /*        ++----------- TOP8 continuous output mode */

#define TOP8_ILevel             0           /* MJT output interrupt 5 (TOP8) interrupt priority
level              */
```

```
135
136 #define TOP8_Cycle              5000 - 1        /* TOP8 interrupt period 5000                    */
137
138 /*""FUNC COMMENT""*************************************************************
139  * Function name : init_func()
140  *-----------------------------------------------------------------------------
141  * Description  : - Initialize ICU
142  *-----------------------------------------------------------------------------
143  * Argument     : -
144  *-----------------------------------------------------------------------------
145  * Returns      : -
146  *-----------------------------------------------------------------------------
147  * Notes        : The timer acts as a first trigger is started at the last
148  *""FUNC COMMENT END""*************************************************/
149 void init_func(void)
150 {
151         port_init();                                    /* Initialize those related to port */
152         dma_init();                                     /* Initialize DMA */
153         sio0tr_init();                                  /* Initialize serial I/O */
154         ad_init();                                      /* Initialize A-D converter  */
155         timer_init();                                   /* Timer initial setting */
156 }
157
158 /*""FUNC COMMENT""*************************************************************
159  * Function name : port_init()
160  *-----------------------------------------------------------------------------
161  * Description  : - Initialize port
162  *-----------------------------------------------------------------------------
163  * Argument     : -
164  *-----------------------------------------------------------------------------
165  * Returns      : -
166  *-----------------------------------------------------------------------------
167  * Notes        : -
168  *""FUNC COMMENT END""*************************************************/
169 void port_init(void)
170 {
171         PICNT = PIEN0;                                  /* Enable port input */
172
173 /*** LED output port ***/
174
175         P11DATA = 0x00;                                 /* Output data (must be set prior to mode) */
176         P11DIR = 0xff;                                  /* P110-P117 :Output mode */
177         P11MOD = 0x00;                                  /* P110-P117 :Input/output port */
178
179 /*** Switch input port ***/
180
181         P13DIR = 0x00;                                  /* P130-P137 :Input mode */
182         P13MOD = 0x00;                                  /* P130-P137 :Input/output port */
183 }
184
185 /*""FUNC COMMENT""*************************************************************
186  * Function name : dma_init()
187  *-----------------------------------------------------------------------------
188  * Description  : - Initialize DMAC
189  *              :   DMA0: Transfer A-D0ch0 conversion result to S100
190  *              :   transmit buffer (infinite)
191  *-----------------------------------------------------------------------------
192  * Argument     : -
193  *-----------------------------------------------------------------------------
194  * Returns      : -
195  *-----------------------------------------------------------------------------
196  * Notes        : -
197  *""FUNC COMMENT END""*************************************************/
198 void dma_init(void)
199 {
200 /*** DMA0 initial setting ***/
201
202         DM04ITMK |= DMITMK0;                            /* Inhibit DMA0 interrupt */
203         DM04ITST = (~DMITST0) & 0xFFu;                  /* Clear DMA0 interrupt request */
204
205         DM0SA = (USHORT)&AD08DT0;                       /* Source address -> A-D convert ch0 8-bit data
*/
206         DM0DA = (USHORT)&S0TXB_L;                       /* Destination address -> SIO0 transmit buffer
*/
207
208         DM0CNT = DMA0_INIT;                              /* Start upon the completion of A-D0 conversion,
```

```
enable ring buffer mode and transfer */
  209  }
  210
  211  /*""FUNC COMMENT""*************************************************
  212   * Function name : sio0tr_init()
  213   *-----------------------------------------------------------------
  214   * Description  : - Set 8-bit UART transmission for SIO0
  215   *-----------------------------------------------------------------
  216   * Argument     : -
  217   *-----------------------------------------------------------------
  218   * Returns      : -
  219   *-----------------------------------------------------------------
  220   * Notes        : No reception setting has done
  221   *              : For M32R/E#1,2,3, PnMOD cannot be accessed for R/M/W
  222   *              : The function must be executed while interrupt is inhibited
  223   *""FUNC COMMENT END""*********************************************/
  224  void sio0tr_init(void)
  225  {
  226  /*** Setting transfer mode */
  227
  228          S0TCNT = SnTCNT_INI;                            /* f(BCLK), disable transmission */
  229          P8MOD |= P8MOD_SCI0;                            /* TXD0 and RXD0 are valid */
  230          S0MOD = SnMOD_INI;                              /* Set data format */
  231          S0BAUR = BAUD_19200;                            /* Set baud rate */
  232
  233  /*** Interrupt related settings ***/
  234
  235          ISIO0TXCR = 7;                                  /* Set SIO0 transmit interrupt priority level
(inhibit) */
  236          SI03SEL &= ~ISR0;                               /* Select reception-finished interrupt */
  237          SI03MASK &= ~T0MASK;                            /* Disable SIO1 transmit interrupt request */
  238
  239  /*** Starting transmission ***/
  240
  241          S0TCNT |= TEN;                                  /* Enable transmission */
  242  }
  243
  244  /*""FUNC COMMENT""*************************************************
  245   * Function name : ad_init()
  246   *-----------------------------------------------------------------
  247   * Description  : - Initialize A-D converter
  248   *-----------------------------------------------------------------
  249   * Argument     : -
  250   *-----------------------------------------------------------------
  251   * Returns      : -
  252   *-----------------------------------------------------------------
  253   * Notes        : -
  254   *""FUNC COMMENT END""*********************************************/
  255  void ad_init(void)
  256  {
  257  /*** Interrupt related settings ***/
  258
  259          IAD0CCR = 7;                                    /* Set AD0 transmit interrupt priority level */
  260
  261  /*** Single mode setting ***/
  262
  263          AD0SIM0 = AD0SIM0_INIT;                         /* Request hardware trigger and DMA0 transfer
*/
  264          AD0SIM1 = AD0SIM1_INIT;                         /* Select double speed, A-D conversion mode and
ch.0 */
  265
  266  /*** Scan mode setting ***/
  267
  268          AD0SCM0 = AD0SCM0_INIT;                         /* Select continuous mode and software trigger
*/
  269          AD0SCM1 = AD0SCM1_INIT;                         /* Double speed, 4-channel scan */
  270
  271          AD0SCM0 |= 0x01u;                               /* Start scan mode A-D conversion */
  272  }
  273
  274  /*""FUNC COMMENT""*************************************************
  275   * Function name : timer_init()
  276   *-----------------------------------------------------------------
  277   * Description  : - Initialize timer
  278   *              :   Count TOP8 in continuous mode for starting AD conversion
  279   *-----------------------------------------------------------------
```

```
  280   * Argument     : -
  281   *-------------------------------------------------------------------
  282   * Returns      : -
  283   *-------------------------------------------------------------------
  284   * Notes        : -
  285   *""FUNC COMMENT END""************************************************/
  286  void timer_init(void)
  287  {
  288          UCHAR   temp;
  289          USHORT  temp16;
  290
  291  /*** Disabling TOP8 (TO8) output ***/
  292
  293          P10MOD &= ~0x80u;                       /* Disable TO8 (TOP8 output) against output .. Select
port P100 output */
  294
  295  /*** Setting disable interrupt ***/
  296
  297          temp = TOPIR3;
  298          temp |= ( TOPIS9 | TOPIM8);                             /* TOP8 Disable interrupt */
  299          TOPIR3 = temp;
  300
  301  /*** Setting continuous output mode(TOP8) ***/
  302
  303          OEBCR &= ~OEB3TOP8;                                     /* Select output event bus 3 for input
-> TOP8 for output */
  304
  305          temp16 = TOP810CR;
  306          TOP810CR = ( temp16 & ~TOP8_MASK) | TOP8_CntOutput;     /* Set TOP8 continuous output mode */
  307
  308          PRS0 = (200 - 1);                                       /* Set prescaler(20us@10MHz) */
  309          TOP8CT = TOP8_Cycle;                                    /* Set initial value in TOP8 counter */
  310          TOP8RL = TOP8_Cycle;                                    /* Set initial value in TOP8 reload
register */
  311
  312          TOPPRO = (~TOP8PRO) & 0xFFFFu;                          /* Enable TOP8 enable protect */
  313          TOPCEN = 0xffff;                                        /* Enable TOP8 count */
  314  }
  315
  316  /*""FUNC COMMENT""*****************************************************
  317   * Function name : main()
  318   *-------------------------------------------------------------------
  319   * Description   : Single mode is forcibly executed by TOP8 during A-D scan
  320   *               : mode operation and the conversion result is outputted from
  321   *               : serial I/O (A-D conversion result is 8-bit).
  322   *               : - A-D0 converter operates in 4-channel scan mode
  323   *               : - Output underflow (100ms@10MHz) of the TOP8 continuous
  324   *               :   output mode to the output event bus 3
  325   *               : - Output event bus 3 triggers A-D0ch0 conversion single mode to start
  326   *               : - DMA0 is started upon the completion of A-D conversion
  327   *               :   (the conversion result is transferred to transmit buffer)
  328   *               : - Transmit serial 0 upon completion of DMA0
  329   *               :   (transferred infinitely in ring buffer mode)
  330   *               : - For A-D conversion result in scan mode,
  331   *               :   the result specified with PORT13 is outputted to LED (PORT11)
  332   *-------------------------------------------------------------------
  333   * Argument     : -
  334   *-------------------------------------------------------------------
  335   * Returns      : -
  336   *-------------------------------------------------------------------
  337   * Notes        : Interrupt is not actually used
  338   *              : A-D conversion result is not ASCII converted
  339   *""FUNC COMMENT END""************************************************/
  340  void main(void)
  341  {
  342  /*** Initialize microcomputer ***/
  343
  344          DisInt();                                               /* Disable interrupt */
  345
  346          init_func();
  347
  348          EnInt();                                                /* Enable interrupt */
  349
  350  /*** Display A-D conversion result on LED ***/
  351
  352          while(1) {
```

```
353                 switch (P13DATA) {
354                         case 0x01:
355                                 P11DATA = AD08DT3;
356                                 break;
357                         case 0x02:
358                                 P11DATA = AD08DT2;
359                                 break;
360                         case 0x04:
361                                 P11DATA = AD08DT1;
362                                 break;
363                         case 0x08:
364                                 P11DATA = AD08DT0;
365                                 break;
366                         default:
367                                 break;
368                 }
369         }
370 }
```

3.4 Timing of operation

Timing of operation in this reference program is shown below.

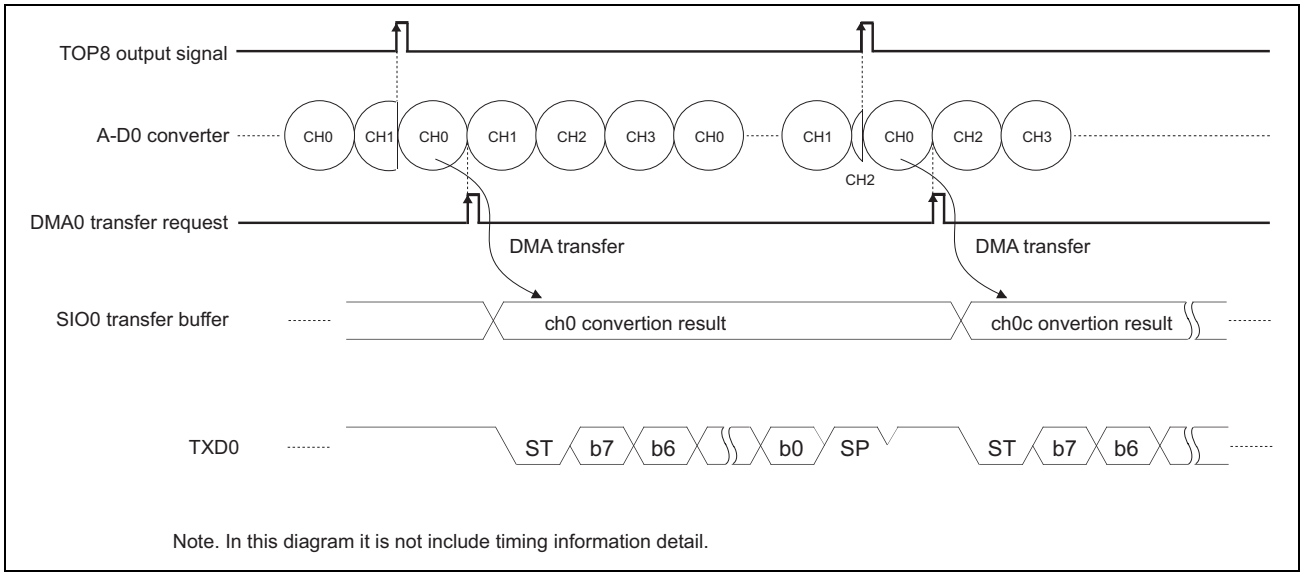


Figure3.4.1 Timing Diagram for combination sample of serial interface transmission and DMAC

4. Reference Documents

- 32176 Group User's Manual (Rev.1.01)
- M32R Family Software Manual (Rev.1.20)
- M3T-CC32R V.4.30 User's Manual (Compiler)
- M3T-CC32R V.4.30 User's Manual (Assembler)

(Please get the latest one from Renesas Technology Corp. website.)

5. Website and Support Center

- Renesas Technology Corp. website

  http://www.renesas.com/

- Inquires for all Renesas products and technical inquiries for the M32R Family products:

  Customer Support Center: csc@renesas.com

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Dec.09.05	—	First edition issued

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.